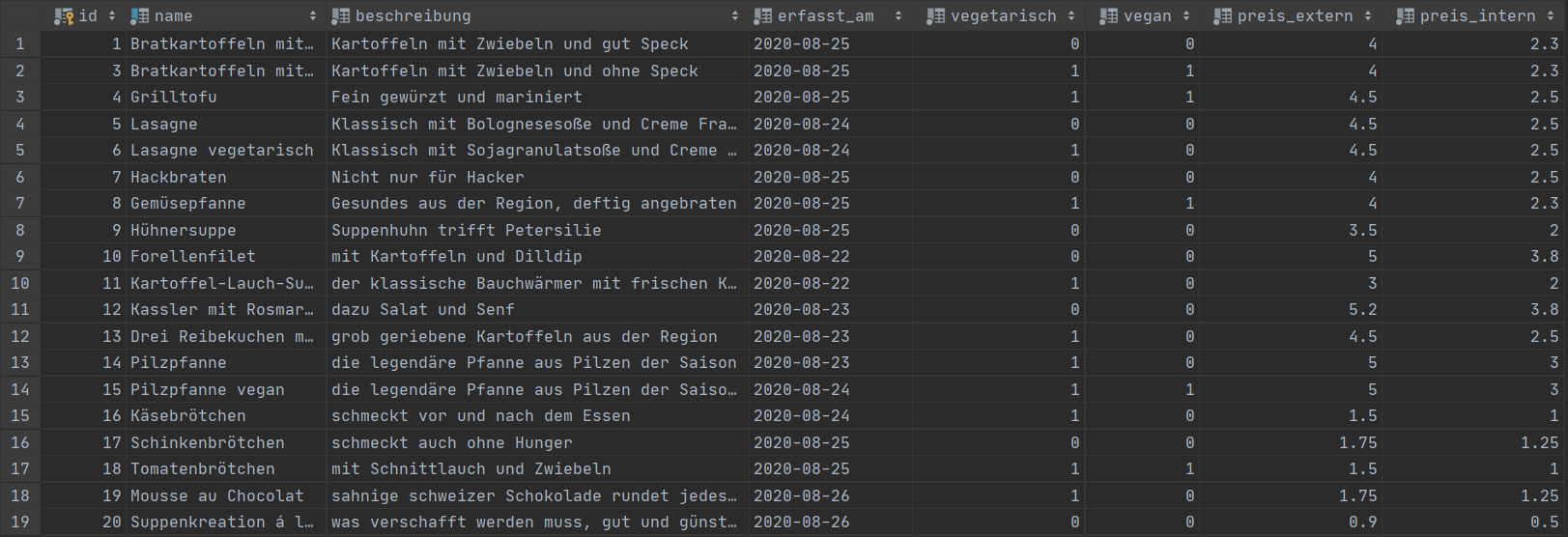
Dossier Meilenstein 3

**Aufgabe 4:**

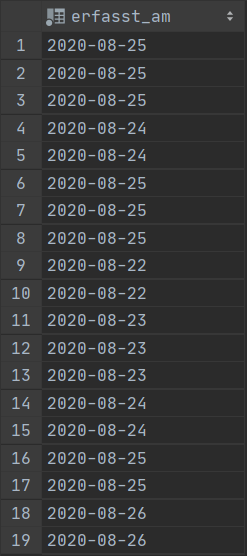
|  |  |
| --- | --- |
| Statement | Beschreibung |
| CREATE DATABASE E\_Mensa; | Datenbank ertsellen |
| ALTER DATABASE test CHARACTER SET='utf8mb4' COLLATE='utf8mb4\_unicode\_ci'; | Zeichensatz festlegen |
| CREATE TABLE IF NOT EXISTS … | Tabellen erzeugen |
| ALTER TABLE gericht DROP CONSTRAINT IF EXISTS |  |
| alter table gericht\_hat\_allergen drop foreign key |  |
| ALTER TABLE gericht\_hat\_allergen  ADD CONSTRAINT gericht\_hat\_allergen\_allergen\_allergencode\_fk  FOREIGN KEY (code)  REFERENCES allergen(code)  ON UPDATE CASCADE  ON DELETE RESTRICT; | Beziehungen zwischen mehreren Tabellen zu definieren  Foreign Key: Spaltenset als Fremdschlüssel verwendet  ON UPDATE CASCADE: automatisches aktualisieren der Schlüssel  ON DELETE RESTRICT: Löschen wird verhindert |
| SELECT COUNT(\*) FROM gericht AS cntG; | Zählt alles Eintrage von Tabelle gericht. |

1) Alle Daten aller Gerichte. Statement: SELECT \* FROM gericht;



2) Das Erfassungsdatum aller Gerichte.

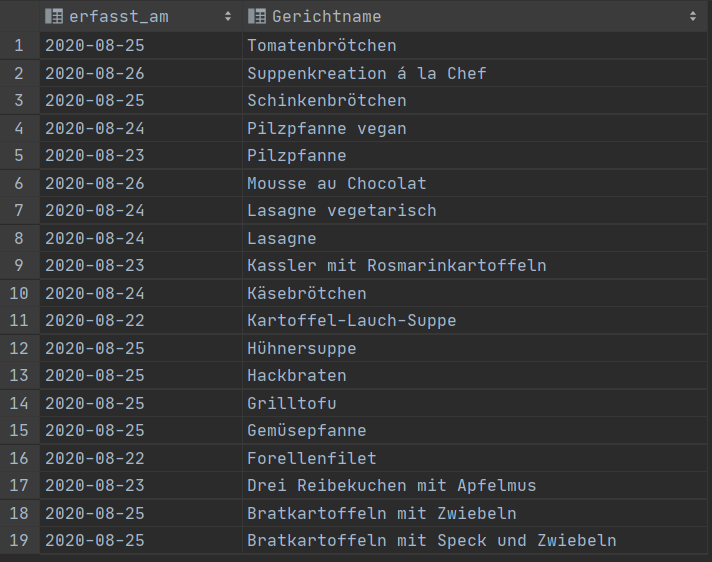
Statement: SELECT erfasst\_am FROM gericht ;



3) Das Erfassungsdatum sowie den Namen (als Attributname Gerichtname) aller

Gerichte absteigend sortiert nach Gerichtname.

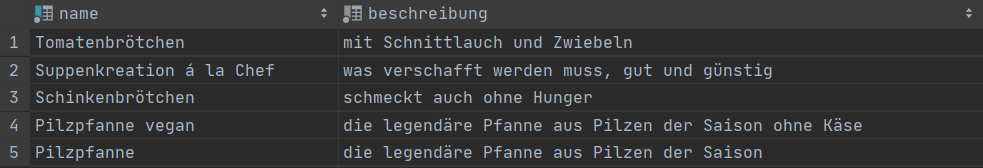
Statement: SELECT erfasst\_am ,name AS 'Gerichtname' FROM gericht ORDER BY name DESC ;



4) Den Namen sowie die Beschreibung der Gerichte aufsteigend sortiert nach

Name, wobei nur 5 Datensätze dargestellt werden sollen.

Statement: SELECT name, beschreibung FROM gericht ORDER BY name DESC LIMIT 5;

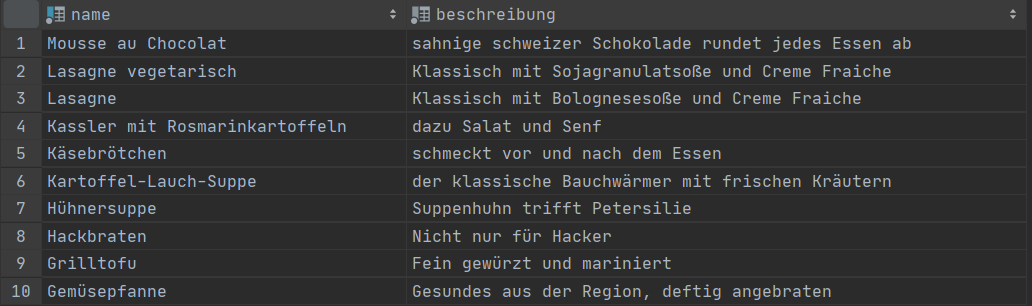


5) Ändern Sie die vorherige Abfrage so ab, so dass 10 Datensätze dargestellt

werden, die nach den ersten 5 Datensätzen folgen. (Die ersten 5 Datensätze

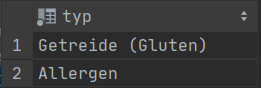
werden übersprungen)

Statement: SELECT name, beschreibung FROM gericht ORDER BY name DESC LIMIT 10 OFFSET 5;



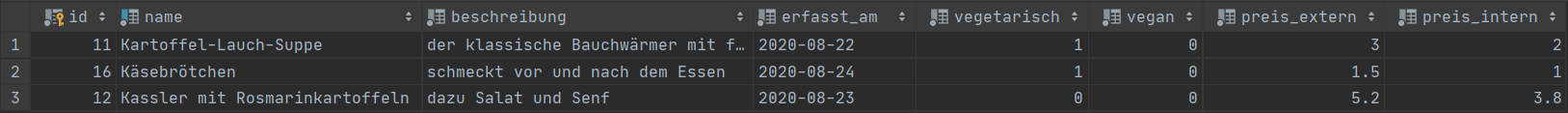
6) Zeigen Sie alle möglichen Allergen-Typen (typ), wobei Sie keine doppelten

Einträge darstellen. Statement: SELECT DISTINCT typ FROM allergen;



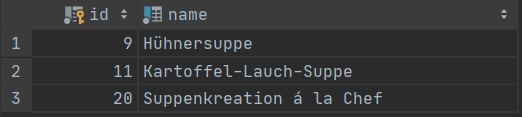
7) Namen von Gerichten, deren Name mit einem „K“ beginnt.

Statement: SELECT \* FROM gericht WHERE name LIKE 'K%';



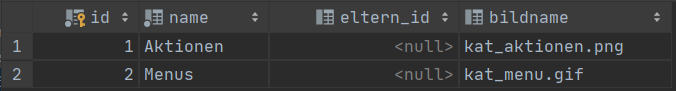
8) Ids und Namen von Gerichten, deren Namen ein „suppe“ an beliebiger Stelle

enthält. Statement: SELECT id,name FROM gericht WHERE name LIKE '%suppe%';



9) Alle Kategorien, die keine Elterneinträge besitzen.

Statement: SELECT \* FROM kategorie WHERE eltern\_id IS NULL ;



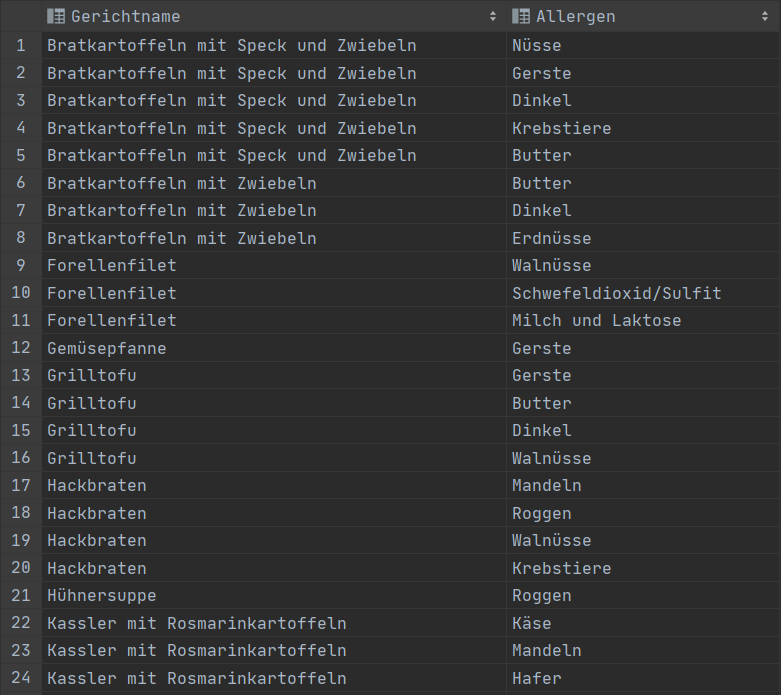
10)Alle Gerichte mit allen zugehörigen Allergenen.

Statement:

SELECT gericht.name AS Gerichtname, a.name AS Allergen FROM gericht

JOIN gericht\_hat\_allergen gha on gericht.id = gha.gericht\_id

JOIN allergen a on a.code = gha.code;



…..

11)Ändern Sie die vorherige Abfrage so ab, dass alle existierenden Gerichte

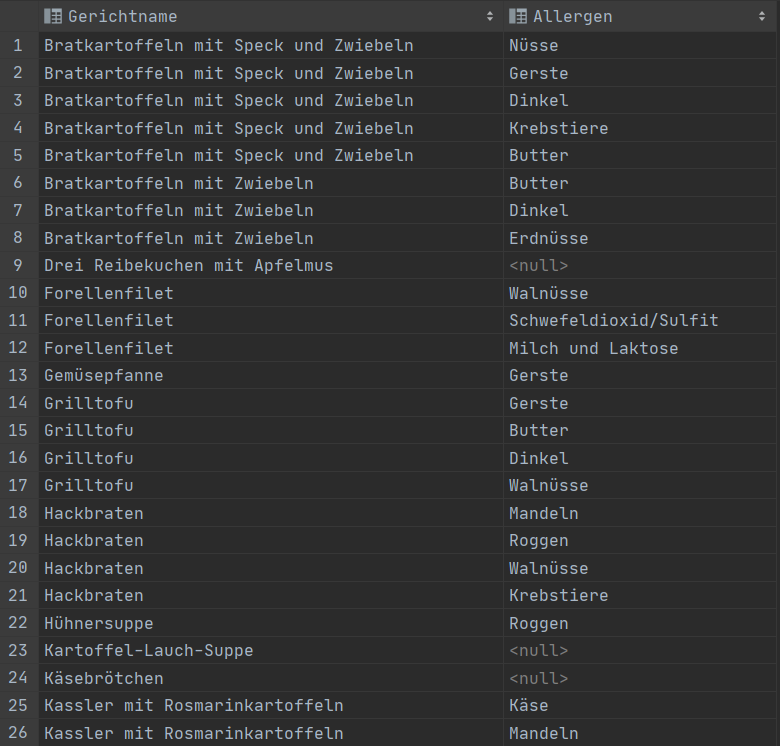
dargestellt werden (auch wenn keine Allergene enthalten sind).

Statement:

SELECT gericht.name AS Gerichtname, a.name AS Allergen FROM gericht

Left JOIN gericht\_hat\_allergen gha on gericht.id = gha.gericht\_id

Left JOIN allergen a on a.code = gha.code;



….

12)Ändern Sie die vorherige Abfrage so ab, so dass im Ergebnis alle

existierenden Allergene dargestellt werden (auch wenn diese nicht einem

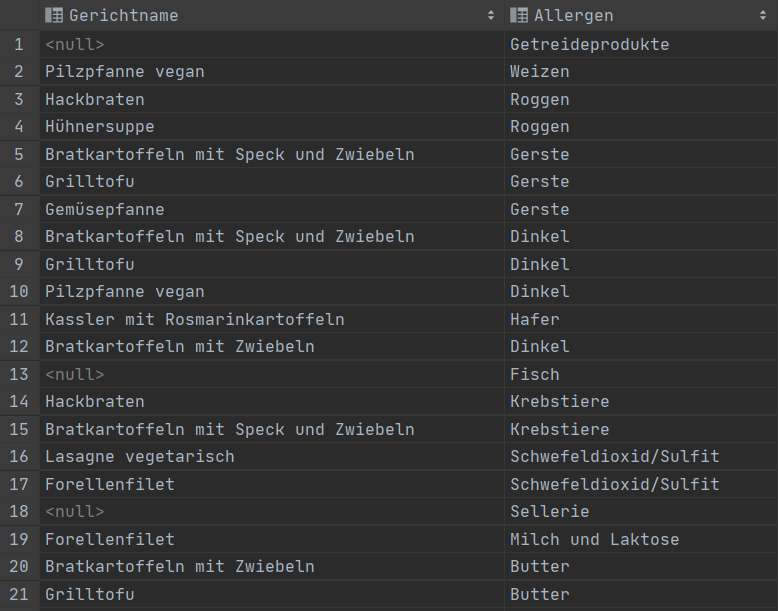
Gericht zugeordnet sind).

Statement:

SELECT gericht.name AS Gerichtname, a.name AS Allergen FROM gericht

Right JOIN gericht\_hat\_allergen gha on gericht.id = gha.gericht\_id

Right JOIN allergen a on a.code = gha.code;



….

13)Die Anzahl der Gerichte pro Kategorie aufsteigend sortiert nach Anzahl.

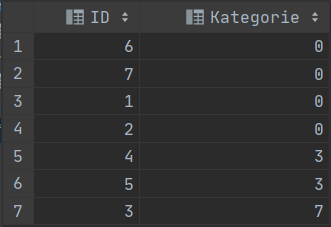
Statement:

SELECT DISTINCT id AS ID,COUNT(Kategorie\_id) AS Kategorie FROM kategorie

Left JOIN gericht\_hat\_kategorie ghk on kategorie.id = ghk.Kategorie\_id

GROUP BY id

ORDER BY Kategorie ASC;



14)Ändern Sie die vorherige Abfrage so ab, dass dabei nur die Kategorien

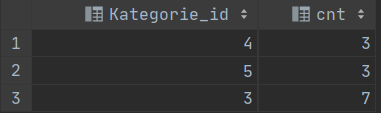
dargestellt werden, die mehr als 2 Gerichte besitzen.

Statement: SELECT DISTINCT id AS ID,COUNT(Kategorie\_id) AS Kategorie FROM kategorie

Left JOIN gericht\_hat\_kategorie ghk on kategorie.id = ghk.Kategorie\_id

GROUP BY id

HAVING Kategorie>2 ORDER BY Kategorie ASC;



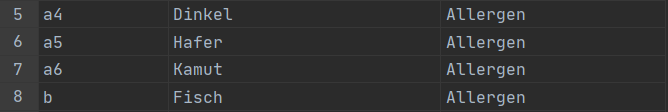
15) Korrigieren Sie den Wert „Dinkel“ in der Tabelle allergen mit dem code a6 zu

„Kamut“.

Statement: UPDATE allergen

SET name = 'Kamut'

WHERE code = 'a6';



16) Fügen Sie das Gericht „Currywurst mit Pommes“ hinzu und tragen Sie es in

der Kategorie „Hauptspeise“ ein.

Statement:

INSERT INTO gericht

VALUES (2,'Currywurst mit Pommes', 'nichts','2020-08-25',0,0,4,2.3);

INSERT INTO gericht\_hat\_kategorie

VALUES (2,3);

